Visual local navigation using warped panoramic images

Binding, Dave; Labrosse, Frédéric

Published in:
Proceedings of Towards Autonomous Robotic Systems 2006

Publication date:
2006

Citation for published version (APA):
Binding, D., & Labrosse, F. (2006). Visual local navigation using warped panoramic images. In M. Witkowski, U. Nehmzow, C. Melhuish, E. Moxey, & A. Ellery (Eds.), *Proceedings of Towards Autonomous Robotic Systems 2006: Incorporating the Autumn Biro-Net Symposium* (pp. 19-26)

Visual local navigation using warped panoramic images

Dave Binding Frédéric Labrosse

Department of Computer Science
University of Wales, Aberystwyth
Aberystwyth SY23 3DB, United Kingdom

e-mail: ffl@aber.ac.uk

Published in: *Proceedings of Towards Autonomous Robotic Systems*, pages 19–26 (2006), University of Surrey, Guildford, United Kingdom, http://www.taros.org.uk.

Visual local navigation using warped panoramic images

Dave Binding **Frédéric Labrosse**
Department of Computer Science, University of Wales
Aberystwyth, Ceredigion, SY23 3ET, UK
ffl@aber.ac.uk

Abstract

In this paper, we present a method that uses panoramic images to perform local homing. Our method is different from others in that it does not extract any features from the images and only performs simple image processing operations. Furthermore, it uses a method borrowed from computer graphics to simulate the effect in the images of translations of the robot to compute local motion parameters.

1 Introduction

Visual navigation increasingly relies on local methods: paths are specified in terms of intermediate targets that need to be reached in succession to perform the navigation task (Vassallo et al., 2002, Neal and Labrosse, 2004, Gourichon, 2004). This task can thus be reduced to a succession of homing steps.

Many homing methods that use vision require the extraction of features from the images and their matching between successive images. This is for example the case for most methods derived from the *snapshot model* (Cartwright and Collett, 1983, Cartwright and Collett, 1987). A review and "genealogy" tree of such methods are given in (Gourichon, 2004) and links between biology and computational models are given in (Ruchti, 2000). A snapshot is a representation of the environment at the homing position, often a one-dimensional black and white image of landmarks and gaps between landmarks, e.g. (Röfer, 1997, Möller et al., 1999), but also two-dimensional images of landmarks such as corners, e.g. (Vardy and Oppacher, 2003). Most of these methods use panoramic snapshots. Although feature extraction can be fast, it often requires assumptions about the type of features being extracted and the world in which the robot is, in particular its structure (Gonzales-Barbosa and Lacroix, 2002). Natural environments often present no obvious visual landmarks or when these exist, they are not necessarily easy to distinguish from their surroundings. The matching of features between successive images is often difficult and also requires many assumptions (see (Gourichon, 2004) for a discussion).

We propose to use the images as they are; this is the *appearance-based* approach (Labrosse, 2006, Mitchell and Labrosse, 2004, Neal and Labrosse, 2004). This is also sometimes called the signal-based approach (Cozman et al., 2000). Using whole two-dimensional images rather than a few landmarks extracted from images reduces aliasing problems; indeed, different places can look similar, especially if "seen" using only a few elements of their appearance.

Not many published papers propose to use raw images; a few examples follow. A one-dimensional panoramic image is used by (Röfer, 1997), from which the optic flow is extracted between successive images to control the robot. An array of light sensitive sensors (typically eight) is used by (Bisset et al., 2003) to represent and recognise places; a process similar to the one described in (Neal and Labrosse, 2004) is used to provide rotation independence. In (Gonzales-Barbosa and Lacroix, 2002), histograms of Gaussian derivative filtered images are used. A detailed study of the pixel-wise comparison (Euclidean distance) between panoramic images captured in outdoors environments is done in (Zeil et al., 2003).

In this paper we address the problem of returning to a homing position from a not too distant starting point in real, unmodified environments. The method uses panoramic images captured using an omni-directional camera (a "normal" camera pointing up at a hyperbolic mirror) and simple image processing as well as algorithms borrowed from computer graphics. Note that a similar method was proposed in (Zeil et al., 2003) but implemented in an unrealistic setting: the robot was not rotating and it was moving along trajectories that were neither efficient nor possible with a mobile robot (the "robot" was a camera mounted on a gantry). Similar work and methods have been presented in (Franz et al., 1998), the important differences being that (1) they use 1D panoramic images and (2) they use features that are "warped" on the 1D ring, while we use complete 2D images without any feature extraction.

Section 2 describes the method used to perform the homing by first describing what the problem is and then our solution to the problem while Section 3 describes the whole homing procedure. Section 4 presents some results. A discussion of the results and a conclusion is provided in Section 5.

2 Method

2.1 The problem

The method relies on the fact that changes in images grabbed by the robot when it moves are progressive and that there is thus a clear mapping between images and the environment of the robot. This mapping however can break in some circumstances, typically when distinct places look similar (i.e. have the same appearance). In the case of homing, this is not a problem because the robot starts from a position that is not too remote from its destination. The mapping is however made more robust by the use of panoramic images (Figure 2 gives some examples and the procedure to obtain these images is detailed in (Labrosse, 2006)). Because such images show everything that is visible from a given view point in one image, less tends to appear and disappear compared to "traditional" (non-panoramic) images when the robot moves. Moreover, because everything is visible, it is more difficult to create two different places having the same appearance.

The problem is to compare images in a way that is both fast and reliable. As mentioned in Section 1, we do not want to use visible features of the environment but rather use simple pixel-wise comparisons of the images. An $h \times w$ pixels image with c colour components per pixel is a point in the *image space*, a space having $h \times w \times c$ dimensions representing all possible images of the given size. Images can thus be compared by measuring the distance between them in that image space. A number of distance metrics are possible and we have tried in previous work an L^1 norm (Manhattan distance) and an L^2 norm (Euclidean distance) not showing any difference in the results obtained by either (Mitchell and Labrosse, 2004). For no other reason than simplicity, continuity and smoothness, the Euclidean distance is used in this work. The distance between two images is thus defined as

$$d(\mathcal{I}_i, \mathcal{I}_j) = \sqrt{\sum_{k=1}^{h \times w} \sum_{l=1}^{c} (\mathcal{I}_j(k,l) - \mathcal{I}_i(k,l))^2}, \qquad (1)$$

where $\mathcal{I}_i(k,l)$ and $\mathcal{I}_j(k,l)$ are the l^{th} colour component of the k^{th} pixel of images $\mathcal{I}_i$ and $\mathcal{I}_j$ respectively. Pixels are enumerated, without loss of generality, in scan-line order from top-left corner to bottom-right corner. We used for this work the RGB (Red Green Blue) colour space, thus having three components per pixel. The combination of Euclidean distance and RGB space is not necessarily the best to use but it is sufficient for our purposes (see (Labrosse, 2006) for a discussion).

Performing such image comparisons is only useful if it can lead to information that is pertinent to the task. Figure 1 shows the distance between a number of grabbed images and a "target" image, in our Lab. The images

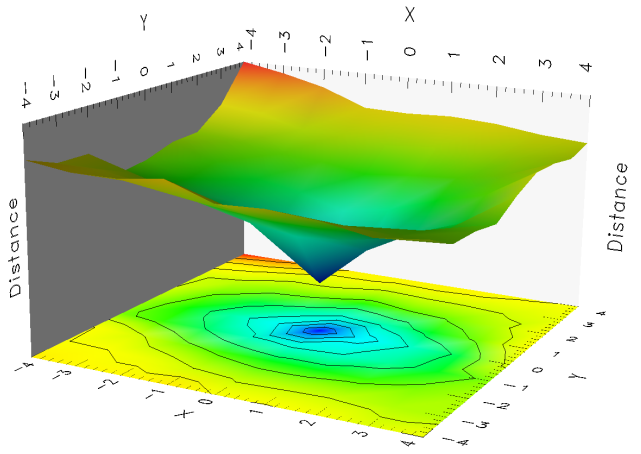

Figure 1: The distance between images grabbed from a number of regular positions and a target image (the coordinates only reflect image index, not coordinates in Cartesian space)

Figure 2: Some of the images used for the experiment on Figure 1: target image (top) and positions $(2, -3)$ (middle) and $(-4, 4)$ (bottom)

were grabbed from positions on a regular grid (81 images in total) and the target image was grabbed from approximately the centre of the grid. The grid size in Cartesian space is approximately 5 m $\times$ 5 m (coordinates on the figure show position on the grid). For all the images the robot was facing in the same direction. Images corresponding to positions on the grid on Figure 1 around $(-4, 4)$ were grabbed while the robot was almost touching large objects (Figure 2, bottom), hence producing larger and faster varying distance values compared to other places where the robot was far from any object. This is consistent with results presented in (Zeil et al., 2003) for an outdoor environment in a large variety of spatial configurations such as far from objects or on the contrary close to objects.

It is clearly visible that the homing position corresponds to the minimum of the distance as a function of space, which in a real situation is not available. It is also clear that a gradient descent on this function will lead to the minimum. This however implies two problems: computing the gradient of the distance function and knowing the heading of the robot relative to the target.

Computing the gradient of the distance function at

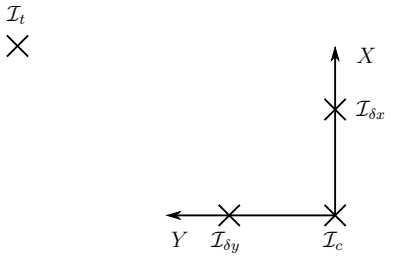

Figure 3: The four images needed to compute the gradient of the distance function at a given position

any point requires four images: the target image $\mathcal{I}_t$, the current image $\mathcal{I}_c$ and two images $\mathcal{I}_{\delta x}$ and $\mathcal{I}_{\delta y}$ taken after two displacements from the current position corresponding to the orthogonal directions X and Y, Figure 3. A solution to that problem was proposed in (Zeil et al., 2003): physically moving the robot to obtain the images $\mathcal{I}_{\delta x}$ and $\mathcal{I}_{\delta y}$. However, in the context of mobile robotics this is not desirable or even possible. It could be argued that only a small triangular trajectory from the current position is needed to obtain the three images. However, such a trajectory is difficult to accurately perform in all but contrived cases and is impossible with many robots (but not with the one we used in this work). Moreover, such a procedure would not be efficient, which is important in many situations.

The problem of knowing the orientation of the robot relative to that of the target can be solved in a number of ways (and as such has been assumed to be available in many papers). One solution is to use an additional sensor such as a magnetic compass. Another is to incorporate the orientation finding in the process. This is possible to some extent for example as part of the feature matching process (Möller et al., 1999) or optical flow computation (Röfer, 1997).

We propose in the next section our solution to these two problems.

2.2 Our solution

The heading of the robot can be estimated using the panoramic images themselves. This is what we use here. For each new image grabbed, the heading is computed by measuring column shifts between the previous images and the current image (Labrosse, 2006). This procedure provides the heading of the robot at each new image with an error within 20° after a long trajectory. For short trajectories involved in homing, the typical error is below 5°. Note that the robot does not need to be physically aligned with the target. This can be simulated by horizontally shifting the images (Labrosse, 2006).

Once the orientation of the robot relative to that of the target is known, the direction of the two translations needed for the gradient computation becomes specified. In this paper we propose to simulate the translations by synthesising the corresponding images from the current image $\mathcal{I}_c$. There is a large number of papers in the literature on image-based rendering in general and simulation of viewpoint in particular. Most of these methods tackle the more general and theoretical problems (Ullman and Basri, 1991, Lu et al., 1998, Tenenbaum, 1998, Shum et al., 2002). In this paper we adopt a more purposive approach because we are only interested in simulating specific short displacements: forward and sideways translations.

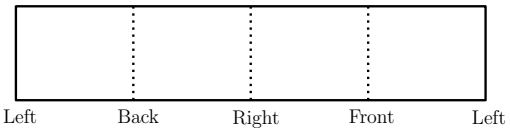

Figure 4: Projection of the environment of the robot onto panoramic images

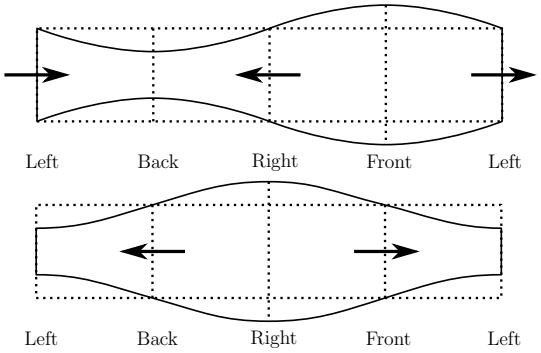

Figure 5: Deformations introduced in panoramic images by forward (top) and sideways to the right (bottom) translation of the robot

The environment of the robot projects onto the panoramic images from left-to-right in the images as the areas corresponding to the left of the robot (column 0), the back (column at 25% of the width), the right (column at 50%), the front (column at 75%) and finally the left again (column at 100%), Figure 4. When the robot moves forward, the part of the image corresponding to the front of the robot expands while the part corresponding to the back contracts. The parts corresponding to the sides move from front to back, Figure 5. A similar deformation happens when the robot moves sideways (to the right). An exact transformation could be computed using the characteristics of the camera if the 3D structure of the environment was available. Indeed, the exact apparent motion of objects in the images depends on their position in the environment relative to the camera. For example, objects close to the camera will move more than objects far away from the camera. Since the 3D structure of the environment is not available only an approximation of the transformation is possible. Moreover, parts of the environment are not visible from the

Figure 6: Three images used to determine the parameters of the warping: $\mathcal{I}_c$ (top), $\mathcal{I}_{\delta x}$ (middle) and $\mathcal{I}_{\delta y}$ (bottom)

Figure 7: The two images simulating the translation respectively corresponding to $\mathcal{I}_{\delta x}$ (top) and $\mathcal{I}_{\delta y}$ (bottom)

current position but should be visible from the translated position. These parts cannot be recovered by any transformation of the images.

We perform the transformation by warping the images using bilinear Coons patches (Heckbert, 1994). The method only needs the boundary of the area of the original images that need to be transformed into the new rectangular image, this for both translations. To obtain these boundaries, the rectangle corresponding to the boundary of the image is transformed as follows:

- the top and bottom edges of the rectangle are regularly sampled into a number of positions (20 in all the experiments reported here);
- each position is shifted horizontally (d_x) and vertically (d_y) according to its position using functions described below;
- the new positions are used as control points for the Coons patches of the top and bottom parts of the boundaries;
- the right and left sides are defined by the extremities of the top and bottom edges and are straight lines.

The two functions are piece-wise polynomials. The degree of the polynomials was determined arbitrarily to be 3 and 1 respectively for the horizontal and vertical shifts of the positions. Other values of the degree have been tried but they have only very limited effect on the performance of the system. The amplitude of the polynomials is more critical as they reflect the amount of simulated translation. These values have been determined experimentally by grabbing three images in a typical homing place corresponding to images $\mathcal{I}_c$, $\mathcal{I}_{\delta x}$ and $\mathcal{I}_{\delta y}$ on Figure 3. An example of these is shown in Figure 6. The centre image $\mathcal{I}_c$ has then been transformed into the images simulating the corresponding translations using the method described above with various values of the amplitudes until images visually similar to the real ones where obtained. These images are shown in Figure 7. Similarity was assessed by comparing the position and size of the visible features in the images in front and on the right of the robot. The retained amplitude values are 9 and 4 respectively for the horizontal and vertical shifts. Figure 8 shows the resulting functions that were used to simulate the forward translations of the robot. The sideways translation functions are the same shifted column-wise by 90. Figure 9 shows the resulting boundaries used by the warping.

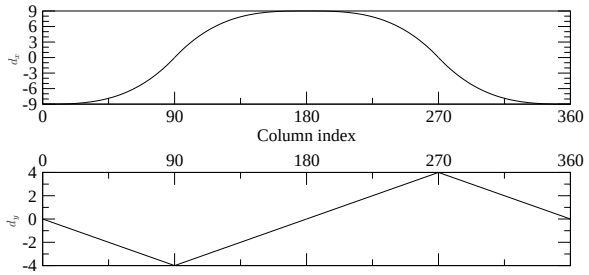

Figure 8: The functions used for the horizontal (top) and vertical (bottom) shifts of positions to obtain the top of the boundary used by the warping to simulate the forward translation. For the bottom part of the boundary, the same functions were used but inverting the column index.

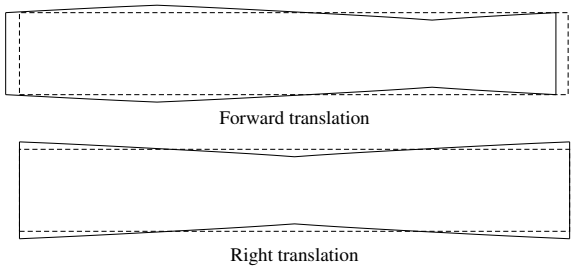

Figure 9: The boundaries used by the warping to simulate the translations (solid) that map onto the image (dashed)

Once the images simulating the translation of the robot are obtained, the gradient of the distance function in image space corresponding to the position of the robot can be computed using the current image $\mathcal{I}_c$, the two simulated images $\mathcal{I}'_{\delta x}$ and $\mathcal{I}'_{\delta y}$ and the target image $\mathcal{I}_t$. Figure 10 shows the gradient computed using the method for each image used to compute the distance function in Figure 1. This clearly shows that the method works in most cases. In particular, the computed gradient is not as good on the top-left corner of the figure. That position corresponds to a place in the environment of the robot which was changing rapidly as a function of the displacement of the robot because of the proximity of large objects, Figure 2 (bottom). This is further

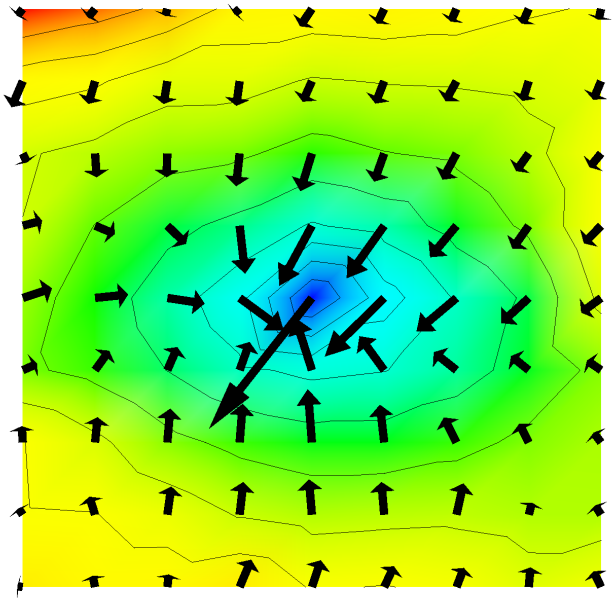

Figure 10: The computed gradient for the images used in Figure 1 using the simulation of the translation

discussed in Section 5. It is interesting to note that the magnitude of the gradient increases dramatically as the robot gets closer to the target.

3 Homing procedure

As previously stated, the heading of the robot relative to that of the target must be known at all times. Using the visual compass (Labrosse, 2006), we must have the heading of the robot at the beginning of the process. This is not too constraining in the context of global navigation where the robot would follow a succession of targets, thus having the possibility of knowing its heading from the previous target.

The algorithm is as follows:

1. Grab a new image $\mathcal{I}_c$.
2. Produce the two images $\mathcal{I}'_{\delta x}$ and $\mathcal{I}'_{\delta y}$.
3. From these images and the target image $\mathcal{I}_t$, compute the gradient at the current position.
4. Use the orientation of the gradient vector to control the heading of the robot and the magnitude of the gradient vector to adjust the forward speed.
5. Repeat 1. until the target is reached.
6. Align the robot with the target.

A simple proportional controller was used for the orientation of the robot. The speed was set to a proportion of the inverse of the gradient magnitude plus one, although in a more general navigation context a constant speed might be better.

Determining when the target is reached can be done in a number of ways. For example, as previously noted, the magnitude of the gradient increases dramatically at the target. A threshold on the magnitude, or its rate of change, or on the distance itself could be used. However, such thresholds are difficult to establish and their value depends on a number of factors such as the pixel values of the images, themselves depending on many factors (illumination, environment, etc.). Instead we use a sudden change in gradient orientation, a method that has been used by others, e.g. (Röfer, 1997). Apart from possibly at the beginning of the process where the robot might not be moving towards the target, the orientation of the gradient should remain constant, or at least not change dramatically until the robot passes the target. However, as can be seen in Figure 10, the gradient at the target can be oriented similarly to the gradient nearby. This results in detecting the target too late or too early depending on where the robot is coming from. As will be seen in the results presented bellow, this contributes to a systematic bias in the homing position.

4 Results

We report here a number of experiments performed in our lab. A motion tracking system VICON 512 was used to track the position and orientation of the robot during the experiments to assess the precision, repeatability and robustness to various parameters. All distances are given in centimetres.

For the first experiment we set up the target position near one side of the play area and started the homing procedure from fifteen different well separated positions. In all cases the robot started with the same orientation as the target. Figure 11 shows the trajectories of the robot for the 15 runs. The orientation of the robot is shown with short thin lines. The target is shown as a star. Apart from two cases, all final positions are in a square 20 cm wide. The centre of the square is 28 cm away from the target. The final orientation is within 5° of the target orientation, although this shows the performance of the visual compass rather than that of the homing procedure. The two failures are due to the fact that the target was close to the limit of the play area with large objects in the top right and left corners in Figure 11. As previously noted this is a situation where the simulation of the translation of the robot does not work well.

The repeatability of the final position of the homing is thus good. However, the accuracy is not as good since the final position is systematically biased even for different targets. Part of the reason is the stopping criterion used, Section 3. The main reason however is probably that the parameters of the warping were not determined with the same images nor the same environment. This is discussed further in Section 5.

The next experiment shows the repeatability of the trajectories. The homing procedure was performed sev-

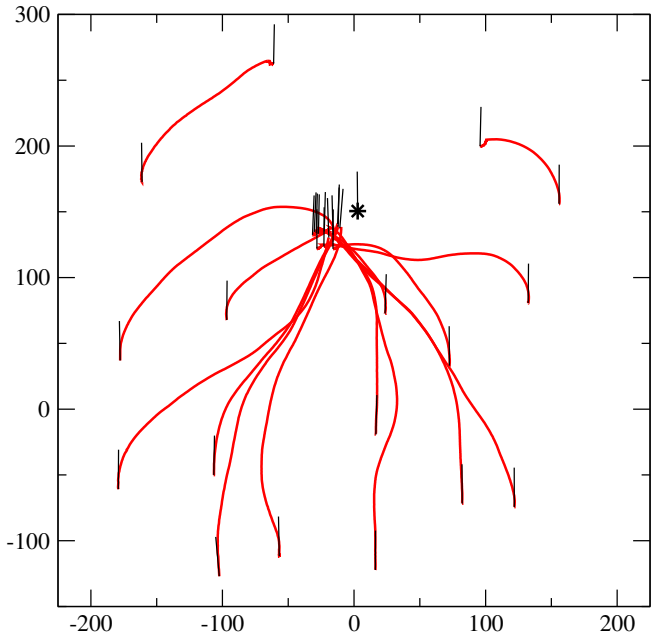

Figure 11: Homing from 15 different initial positions. The initial and final orientations of the robot are show by short thin lines. The target is marked by a star.

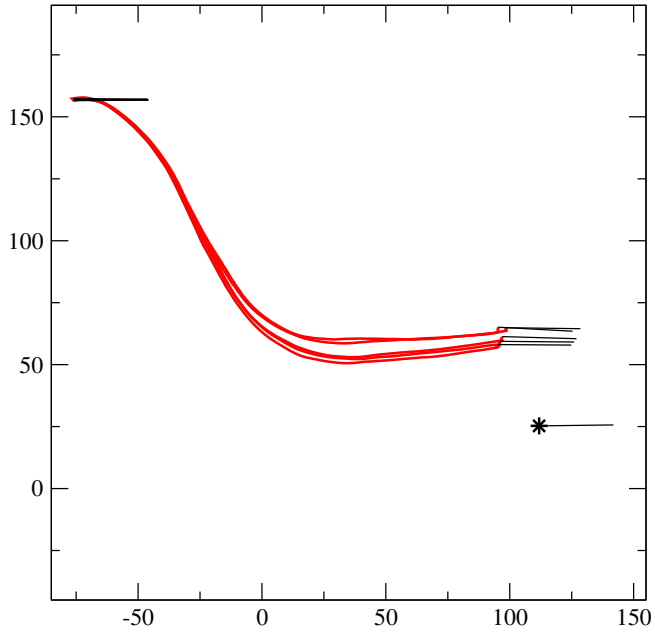

Figure 12: Repeatability of the homing procedure: five runs of the experiment

eral times from the same starting pose (within 5 mm in the x direction, 8 mm in the y direction and 1°). Figure 12 shows the results of the experiment for five trials. All the trajectories finish less than 8 cm from each other and the largest distance between trajectories is about 10 cm. A similar bias as in the previous experiment is observed.

The effect of the heading of the robot at the beginning of the procedure is evaluated in the next experiment. Using the same target as the previous experiment, the homing was started from roughly the same position as before (within 5 cm in the x direction and 13 cm in the y direction, except for one case, see below) but with four different orientations relative to that of the target: +0°, +90°, +180° and +270°, all correctly specified at the beginning of the procedure. For the case of +180°, we had to start from a slightly different position (about 65 cm in the x direction from the other starting positions) because of space constraints. Obviously the trajectories are different, but again good repeatability of the final position is obtained with a bias similar to the previous experiments.

An assumption made in Section 2.2 is that the heading of the robot is known at all times. As previously stated, we used the visual compass that incrementally estimates the change in heading. An initial error in the heading is thus significant for the heading values as the robot progresses. This is tested in the following experiment where the homing procedure was started from the same position with varying orientations while the robot was told that its starting orientation was that of the target. These initial real orientations are given in Table 1 and the resulting trajectories are shown in Figure 14. The three trajectories in dotted line did not finish (were interrupted before the robot stopped). All the others did finish. As expected, the orientation at the end matches that at the beginning (because the visual compass works as expected). The trajectories with an error in heading of less than 4° to the right and 6° to the left still stop at the expected position (with the same bias as in previous experiments). The procedure fails with larger errors. However, the general direction of the trajectory is right and the x position is fairly accurate. This is probably due to a strong visual cue that was present approximately at position $(100, -100)$ and was contributing mostly in translations along the x direction (a long but thin polystyrene sheet standing there to limit the play area).

5 Discussion and conclusion

All the results show good performance of the visual homing procedure proposed here. In particular, very good repeatability has been demonstrated and the method proves more robust than expected when the robot is given erroneous information about its initial heading (the only information required).

The method makes a number of assumptions. The first is that, at least for the final stage of the homing, the robot must be able to turn on the spot. This is because no trajectory planning has been incorporated in the procedure to ensure that the robot arrives at the tar-

Table 1: Wrong initial heading values for the different trajectories of Figure 14

Index	target	0	1	2	3	4	5	6	7	8	9	10
Heading	179	179	188	183	175	173	169	162	149	141	131	113

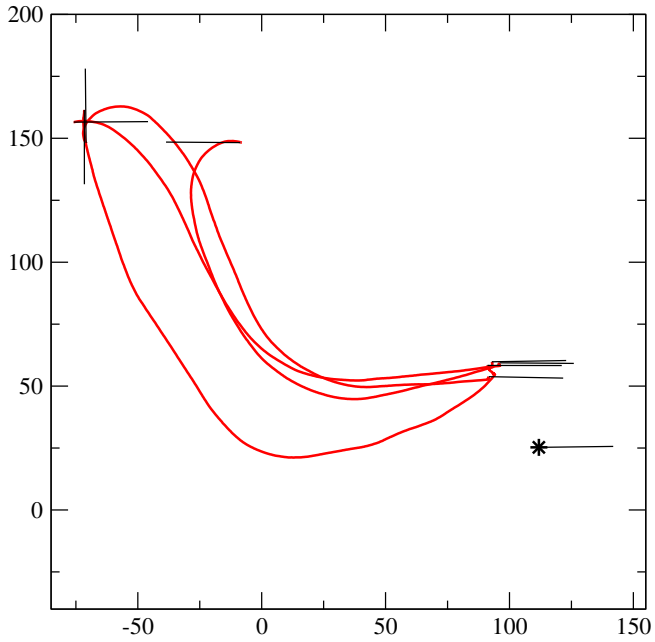

Figure 13: Effect of the orientation of the robot at the beginning of the homing procedure

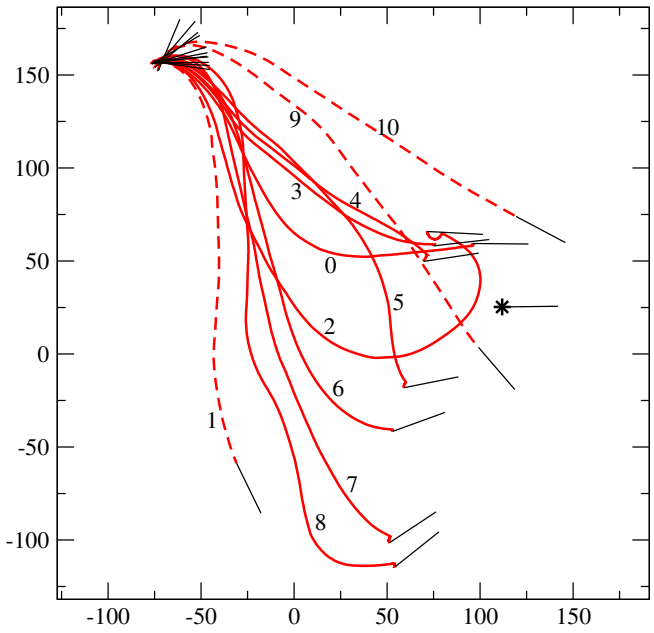

Figure 14: Effect of giving the wrong heading value to the robot at the beginning of the homing procedure (the values are given in Table 1)

get with the correct heading. This is obviously a problem that needs to be solved should the method be applied to a car-like robot. However, in the context of global navigation, i.e. "homing" on a number of successive targets, reaching the correct orientation at each target is not necessarily important and correct orientation could be easily reached for the final target by specifying a number of close sub-targets just before the end of the path.

The second assumption made by the method is that the parameters of the warping simulating the translation of the robot for the gradient computation work in all situations. This is obviously wrong. For example, the apparent motion of the objects in the images is more important for objects that are close to the robot. This is even truer for objects that lie on a line perpendicular to the direction of the translation. In other words, we use the "equal distance assumption" that others have also used (Franz et al., 1998). However, the parameters used in these experiments have been obtained in an environment that was different (objects placed differently around the play area) and for a position that was different than the target positions used here. Despite this, the performance shown is good. However, this might be the reason for the systematically encountered bias. Because targets can be in very different locations for global navigation, establishing parameters on a per target basis might be a good idea and this should be done automatically from real images grabbed after a short (forward) translation. This automatic method would use a minimisation of the distance between the real image and the image simulating the translation. Moreover, in the case of asymmetrical environments different warping parameters should be used to simulate the forward and sideways translations. However, automatically determining the parameters of the sideways translation might be a problem as many robots cannot reliably perform that translation. Finally, the actual directions of the simulated translations should probably be such that they align with visual symmetries of the environment. Such symmetries are however difficult to establish.

The third assumption is that the target image can still be obtained, which is not necessarily the case. For example if the overall illumination changes between the target acquisition and the homing, then it is probable that the performance will decrease. It is however possible to solve that kind of problem by using different colour spaces (Woodland and Labrosse, 2005) and using shadow removal or colour constancy methods.

To conclude, we have described a method that uses simple image comparisons and techniques borrowed from computer graphics to perform visual homing of a robot.

The performance of the method has been evaluated in a number of situations and shown to be good, despite presenting a systematic bias in the final position. More tests need to be performed to assess the catchment area of the method around the target. It is however expected that the result of such a study will be highly dependent on the actual environments, for example producing large catchment areas in wide uncluttered spaces and smaller areas in the presence of many obstacles.

References

Bisset, D. L., Aldred, M. D., and Wiseman, S. J. (2003). Light detection apparatus. United States Patent US 6,590,222 B1. Also UK Patent GB 2 344 884 A, 2000.

Cartwright, B. A. and Collett, T. S. (1983). Landmark learning in bees: experiments and models. *J. of Comparative Physiology*, 151:521–543.

Cartwright, B. A. and Collett, T. S. (1987). Landmark maps for honeybees. *Biol. Cyber.*, 57(1/2):85–93.

Cozman, F., Krotkov, E., and Guestrin, C. (2000). Outdoor visual position estimation for planetary rovers. *Auton. Robots*, 9(2):135–150.

Franz, M. O., Schölkopf, B., Mallot, H. A., and Bülthoff, H. H. (1998). Where did I take that snapshot? Scene-based homing by image matching. *Biol. Cyber.*, 79:191–202.

Gaspar, J., Winters, N., and Santos-Victor, J. (2000). Vision-based navigation and environmental representations with an omnidirectional camera. *IEEE Trans. on Robot. Autom.*, 16(6):890–898.

Gonzales-Barbosa, J.-J. and Lacroix, S. (2002). Rover localization in natural environments by indexing panoramic images. In *Proc. IEEE Int. Conf. in Robot. and Autom.*, pages 1365–1370, Washington, USA.

Gourichon, S. (2004). *Utilisation d'un compas visuel pour la navigation d'un robot mobile*. PhD thesis, Université Paris VI, Paris, France.

Heckbert, P. S. (1994). Bilinear Coons patch image warping. In *Graph. Gems IV*, pages 438–446. Academic Press.

Jogan, M. and Leonardis, A. (2000). Robust localization using panoramic view-based recognition. In *Proc. of the Int. Conf. on Pat. Rec.*, volume 4, pages 136–139, Barcelona, Spain.

Labrosse, F. (2006). Appearance-based heading estimation: the visual compass. Technical Report UWA-DCS-06-048, Computer Science Department, University of Wales, Aberystwyth, UK.

Lu, H.-m., Fainman, Y., and Hecht-Nielsen, R. (1998). Image manifolds. In *Proc. of SPIE; Appl. of Artif. Neural Networks in Image Proc. III*, volume 3307, pages 52–63, San Jose, CA, USA.

Mitchell, T. and Labrosse, F. (2004). Visual homing: a purely appearance-based approach. In *Proc. of Towards Auton. Robotic Sys.*, pages 101–108, University of Essex, Colchester, UK.

Möller, R., Maris, M., and Lambrinos, D. (1999). A neural model of landmark navigation in insects. *Neurocomputing*, 26–27:801–808.

Neal, M. and Labrosse, F. (2004). Rotation-invariant appearance based maps for robot navigation using an artificial immune network algorithm. In *Proc. of the Cong. on Evol. Comput.*, volume 1, pages 863–870, Portland, Oregon, USA.

Röfer, T. (1997). Controlling a wheelchair with image-based homing. In *Proc. of the AISB Symp. on Spatial Reason. in Mobile Robots and Animals*, Manchester University, UK.

Ruchti, S. (2000). Landmark and compass reference in landmark navigation. Master's thesis, Artificial Intelligence Lab des Institutes für Informatik der Universität Zürich, Germany.

Shum, H.-Y., Wang, L., Chai, J.-X., and Tong, X. (2002). Rendering by manifold hopping. *Int. J. of Computer Vision*, 50(2):185–201.

Tenenbaum, J. B. (1998). Mapping a manifold of perceptual observations. *Advances in Neural Info. Process. Systems*, (10).

Ullman, S. and Basri, R. (1991). Recognition by linear combination of models. *IEEE Trans. on Pat. Anal. and Machine Intel.*, 13(10):992–1006.

Vardy, A. and Oppacher, F. (2003). Low-level visual homing. In *Advances in Artificial Life: Proc. of the Europ. Conf. on Art. Life*.

Vassallo, R. F., Santos-Victor, J., and Schneebeli, H. J. (2002). Using motor representations for topological mapping and navigation. In *Proc. of the Int. Conf. on Intel. Robots and Systems*, pages 478–483, Lausanne, Switzerland.

Woodland, A. and Labrosse, F. (2005). On the separation of luminance from colour in images. In *Proc. of the Int. Conf. on Vision, Video, and Graphics*, pages 29–36, University of Edinburgh, UK.

Zeil, J., Hofmann, M. I., and Chahl, J. S. (2003). Catchment areas of panoramic snapshots in outdoor scenes. *J. of the Optic. Soc. of America A*, 20(3):450–469.